

VŠB – Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

Katedra kybernetiky a biomedicínského inženýrství

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student: **Jindřich Brablík**
Studijní program: B2649 Elektrotechnika
Studijní obor: 3901R039 Biomedicínský technik
Téma: **Absolvování individuální odborné praxe**
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: ELCOM, a.s.
2. 2. Struktura závěrečné zprávy:
 - a. Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b. Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c. Zvolený postup řešení zadaných úkolů.
 - d. Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e. Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f. Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vedl odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **doc. RNDr. Jindřich Černohorský, CSc.**

Konzultant bakalářské práce: Ing. Jindřich Kaluža

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015

doc. Ing. Jiří Koziolek, Ph.D.
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 7. 5. 2015



.....
Jindřich Brablík

Prohlášení zástupce spolupracující právnické nebo fyzické osoby

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských/magisterských programech VŠB-TU Ostrava.

V Ostravě dne 7. 5. 2015



Ing. Jindřich Kaluža

Poděkování

Rád bych poděkoval kolegům Ing. Adamu Kačmaříkovi, Ing. Gustavu Hrudkovi, Phd a Ing. Jiřímí Pospíšilíkovi, Phd z oddělení zkušeben, za věnovaný čas, podporu a předané znalosti. Mé poděkování také patří pracovníkům vedení firmy ELCOM, a.s. za umožnění praxe a projevenou důvěru při realizaci projektu. V neposlední řadě bych chtěl poděkovat mému internímu vedoucímu doc. RNDr. Jindřichu Černohorskému a konzultantovi na firmě Ing. Jindřichu Kalužovi za odborné vedení, ochotu a trpělivost kterou mi v průběhu praxe věnovali.

Abstrakt

Tato práce je zprávou o mnou absolvované praxi v divizi virtuální instrumentace firmy ELCOM, a.s. Hlavní náplní odborné praxe byl návrh a realizace softwarových aplikací tří vrstev pro tři různé hardwarové platformy (hradlové pole, PC a RTOS mikroprocesor) přístroje pro měření kapacity impregnovaných statorových svazků asynchronních motorů. Součástí praxe bylo také pochopení konceptu a účelu realizovaného přístroje, seznámení se s jeho jednotlivými komponenty a spolupráce s lidmi z různých oddělení divize. Firma ELCOM, a.s. využívá převážně produktů firmy National Instruments, proto byl navržený software implementován ve vývojovém prostředí LabVIEW a jeho přidružených modulech.

Klíčová slova

ELCOM, a.s.; Virtuální instrumentace; Tester; National instruments; LabVIEW; Hradlové pole; Mikroprocesor s operačním systémem reálného času; Měření a sběr dat

Abstract

This bachelor thesis is a report from technical internship which I have attended at ELCOM, a.s. company, specifically its division of virtual instrumentation. Main subject matter of my internship was to design and implement software applications of three layers for three different hardware platforms (Field programmable gate array, PC, RTOS microprocessor) intended for device for measuring capacitance of impregnated stator winding of asynchronous motors. Other part of this internship was to understand the concept and purpose of the device, study its individual components and cooperate with co-workers from various department of the division. ELCOM, a.s. uses mainly products from National Instruments, this is a reason why the designed software was implemented in integrated development environment LabVIEW and its expanding modules.

Key words

ELCOM, a.s.; Virtual instrumentation; Tester; National Instruments; LabVIEW; Field programmable gate array; Real-time OS microprocessor; Measurement and data acquisition

Seznam zkratek

Zkratka	Význam	Český význam
FPGA	Field Programmable Gate Array	Programovatelné hradlové pole
RIO	Reconfigurable Input Output	Nastavitelné vstupy a výstupy
PMU	Power Management unit	Jednotka správy napájení
DIO	Digital Input Output	Digitální vstupy a výstupy
LED	Light Emitting Diode	Dioda emitující světlo
NI	National Instruments	National Instruments
SW	Software	Software
HW	Hardware	Hardware
MAX	Measurement and Automation Explorer	Průzkumník pro měření a automatizaci
FXP	Fixed Point data type	Datový typ s pevnou desetinou čárkou
VISA	Virtual instrumentation software architecture	Softwarová architektura pro virtuální instrumentaci
OLED	Organic Light Emitting Diode	Organická dioda emitující světlo
A/D	Analog to Digital	Analogově digitální (převodník)
FIFO	First In First Out	Paměť typu první dovnitř, první ven
RTOS	Real Time Operating System	Operační systém reálného času
THD	Total Harminic Distortion	Celkové harmonické zkreslení
FFT	Fast Fourier Transform	Rychlá Fourierova transformace
PSP	Publish and Subscribe Protocol	Komunikační protokol firmy NI (publikuj a odebírej)
TCP	Transmition Control Protocol	Protokol ovládání přenosu dat

Obsah

Úvod.....	10
1 Historie a informace o firmě ELCOM, a. s.....	11
1.1 Odborné zaměření firmy.....	11
1.2 Divize Virtuální instrumentace.....	12
2 Zadání.....	13
2.1 Popis funkce přístroje.....	13
2.2 Zadané úkoly.....	14
3 Postup řešení zadaných úkolů.....	16
3.1 Seznámení se s vestavěným systémem a potřebným SW.....	16
3.1.1 Technické parametry vestavěného systému NI sbRIO 9626.....	17
3.1.2 SW potřebný k nastavení a programování NI sbRIO 9626.....	17
3.2 Generování excitačního signálu.....	18
3.3 Oživení Čtečky čárových kódů.....	19
3.3.1 Konfigurace datového přenosu a realizace komunikace.....	20
3.4 Oživení Displeje.....	21
3.5 Sensor pro měření teploty.....	22
3.6 Vzorkování měřených signálů.....	23
3.7 Softwarová aplikace pro hradlové pole.....	24
3.8 SW aplikace pro RTOS mikroprocesor.....	26
3.8.1 Přenos dat mezi aplikací hradlového pole.....	26
3.8.2 Vyhodnocení měřených dat.....	26
3.8.3 Kalibrace zařízení.....	27
3.8.4 Realizace algoritmů pro diagnostiku měřených dat.....	27
3.8.5 Realizace algoritmů pro práci se soubory a daty.....	28
3.8.6 Návrh uživatelského prostředí a ovládání přístroje.....	28
3.8.7 Komunikace s jednotkou správy napájení.....	29
3.8.8 Komunikace s aplikací nadřazené zkušebny.....	29

3.9	SW aplikace pro PC nadřazené zkušebny.....	30
3.10	Optimalizace a implementace vytvořeného softwaru	30
3.11	Předání přístroje zákazníkovi.....	31
4	Teoretické a praktické znalosti získané studiem uplatněné v průběhu praxe	32
5	Scházející znalosti a dovednosti v průběhu praxe.....	33
6	Závěr a zhodnocení praxe	34
	Seznam použité literatury a zdrojů	35
	Seznam obrázků	36

Úvod

Měření, řízení a zpracování dat je podstatnou součástí téměř všech zdravotnických zařízení pro diagnostiku i terapii. Oddělení automatizovaných zkušeben divize virtuální instrumentace firmy ELCOM, a. s. se problematikou měření a řízení zabývá. Hlavním zaměřením oddělení je návrh a realizace testerů a zkušeben pro zařízení silnoproudého elektrotechnického průmyslu. Součástí dodávaných zařízení je i řídicí a monitorovací software, který vyvíjí přímo zaměstnanci tohoto oddělení.

Odbornou praxi v tomto oddělení jsem si vybral právě proto, abych si osvojil problematiku měření a řízení. Dále jsem chtěl uplatnit své vědomosti získané během studia při práci na reálném projektu. Výběr pracovní pozice pro mou praxi také velmi ovlivnila možnost programovat hradlové pole pomocí vývojového prostředí LabVIEW (Laboratory Virtual Instrument Engineering Workbench), které se v oddělení zkušeben využívá pro vývoj software. Měl jsem tak možnost využít a prohloubit své znalosti tohoto vývojového prostředí, které jsem nabyl studiem předmětů virtuální instrumentace.

I když se oddělení zkušeben zabývá odlišným průmyslovým odvětvím, znalosti nabyté při vykonávání této praxe jsou přenosné a je možné je aplikovat i při návrhu a tvorbě zdravotnických zařízení. Zejména znalosti problematiky v oblasti měření, řízení, zpracování dat a také zkušenosti s návrhem, implementací a zaváděním softwaru do reálných projektů. Pro obor biomedicínské techniky jsou užitečné i znalosti vývojového prostředí LabVIEW, jelikož obsahuje knihovny pro generování a zpracování biosignálů. Je proto vhodné pro vývoj softwaru zdravotnických zařízení.

1 Historie a informace o firmě ELCOM, a. s

Firma ELCOM byla založena v roce 1990 jako společnost s ručením omezeným. Její zakladatelé jsou Ing. Jiří Holoubek a Ing. Vladimír Korenc.

Původní záměr obou inženýrů byl vytvořit Inženýrské a konzultační středisko v oboru silnoproudé elektrotechniky, specializované na oblast kompenzace jalového výkonu, elektromagnetické kompatibility a optimalizaci spotřeby elektřiny. Vysoká poptávka průmyslu po kvalitních službách a hlavně po výrobcích v tomto oboru, vytvořily základ pro postupné rozšíření společnosti o další oblasti.

V roce 1992 byly aktivity společnosti rozšířeny o oblast střídavých regulovaných pohonů a speciálních elektromotorů. V tomtéž roce společnost začala provádět i vlastní elektromontáže a také zahájila i vlastní dílenskou výrobu v pronajatých prostorách podniku MEZ Nedvědice. O rok později pak byla ve společnosti vytvořena i projektová kancelář na nově zřízeném pracovišti v Ostravě. V té době již měla společnost pokryta čtyři místa na mapě České republiky a to v Praze, Brně, Nedvědicích a Ostravě. Počet zaměstnanců a finanční obrát společnosti průběžně narůstá. V roce 1996 byly vytvořeny podmínky na transformaci původní společnosti s ručením omezeným na akciovou společnost.

Úspěšný projekt analyzátoru energetických rušení BK500 byl základem spolupráce s katedrou měření a řízení na TU VŠB FEI v Ostravě. Tato spolupráce vyústila v roce 1997 k založení významné divize Virtuální instrumentace, ve které se projekt BK dodnes i nadále úspěšně rozvíjí.

V roce 1998 firma zakoupila továrnu ve městě Bystřice nad Pernštejnem, kde poté soustředila kompletní elektrotechnickou výrobu společnosti. Dva roky nato se struktura korporace ustálila na pěti divizích, ve kterých pracuje kolem sto padesáti zaměstnanců. V současné době má společnost i zahraniční pracoviště v Abu Dhabi, Hong Kongu a Changshu. [1]

1.1 Odborné zaměření firmy

Firma ELCOM, a. s. podniká v oblasti silnoproudé elektrotechniky, elektroenergetiky, virtuální instrumentace a je organizačně rozdělena do pěti divizí a útvarů zajišťujících obchodování, organizační správu a samotné účetnictví firmy. Místem konání mé praxe byla divize Virtuální instrumentace. [2]

1.2 Divize Virtuální instrumentace

Divize Virtuální instrumentace působí od roku 1997 ve Vědeckotechnologickém parku Ostrava v těsném sousedství areálu VŠB - Technické univerzity Ostrava. Za dobu své existence rozvinula řešení projektů systémové integrace do té míry, že je dnes světově uznávaným dodavatelem zákaznických řešení zejména v oblasti automatizovaných měřicích a testovacích systémů, systémů monitoringu kvality elektřiny, průmyslových testerů a kamerových systémů. Dynamický rozvoj divize virtuální instrumentace byl završen v roce 2006 stavbou moderního technologického centra firmy ELCOM, a.s., v němž dnes divize provozuje a rozvíjí své aktivity.[2]



Obr. 1 Budova divize virtuální instrumentace firmy ELCOM, a.s. [2]

2 Zadání

Náplní mé odborné praxe byla tvorba softwarových aplikací ve vývojovém prostředí LabVIEW a jeho přidružených modulech (Real-time, FPGA). Aplikace byly určených pro přenosný přístroj sloužící k měření kapacity impregnovaných svazků vinutí asynchronních motorů (dále jen tester). Tento tester se skládá z několika základních komponent, jejichž funkce bylo potřebné spojit do funkčního celku prostřednictvím již zmíněných softwarových aplikací.

2.1 Popis funkce přístroje

Přístroj měl umožňovat synchronizaci dat s aplikací nadřazené zkušebny. Synchronizace zajišťovala načtení dat potřebných pro měření svazků a odesílání dat s vyhodnocenými výsledky. Identifikace měřených svazků (zakázek) měla být provedena načtením kódu svazku pomocí čtečky čárových kódů. Pro korektní vyhodnocení kapacity svazku bylo potřebné, aby tester umožňoval okamžité měření teploty.

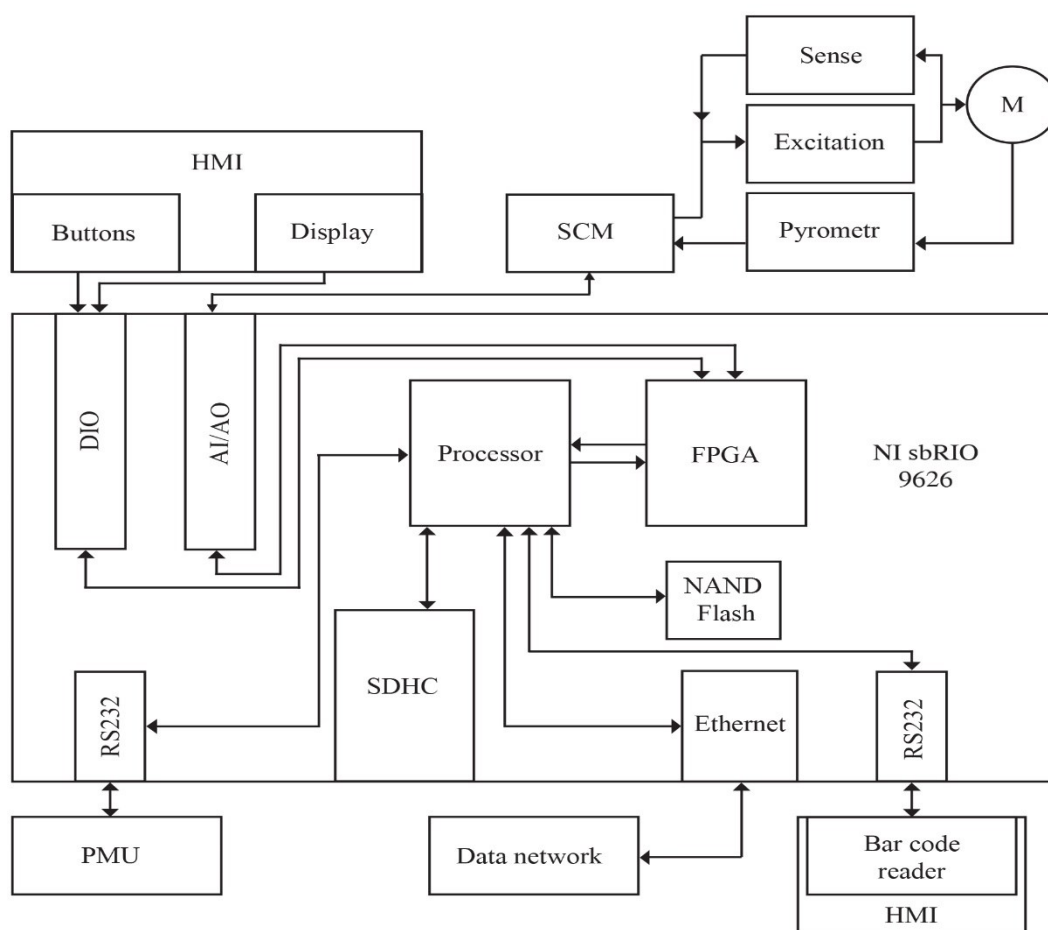
Tester dále zabezpečoval generování excitačního signálu a měření signálů potřebných pro určení kapacity svazku. Hodnoty získané měřením měl přístroj analyzovat, vyhodnotit podle načtených vstupních dat a uložit do výstupního souboru, jehož formát byl specifikovaný zákazníkem.

Hlavním prvkem celého testeru je produkt celosvětově známe firmy National Instruments zabývající se virtuální instrumentací. V současné době je v této oblasti vedoucí firmou, díky svému širokému portfoliu hardwarových a softwarových produktů. Zvoleným produktem je vestavěný (embedded) systém Single board RIO.

Mezi další použité komponenty patří displej firmy Midas Components a čtečka čárových kódů firmy SICK. Pro měření teploty byl zvolen pyrometr firmy MICRO-EPSILON.

Součástí testeru je také bateriová jednotka PMU (Power management unit), zajišťující správu napájení. V případě odpojení přístroje od zdroje elektrické energie, je schopna jej sama napájet po dobu několika hodin. Bateriová jednotka obsahuje mikroprocesor řídící správu napájení a také komunikaci s hlavní softwarovou aplikací testeru.

Poslední součástí testeru je hardwarová deska s tlačítky připojenými na DIO (digital input output) linky vestavěného systému, která spolu s displejem tvoří rozhraní pro ovládání testeru. Schématický náčrt testeru je zobrazen na obr. 2.



Obr. 2 Přehledové schéma realizovaného testeru

2.2 Zadané úkoly

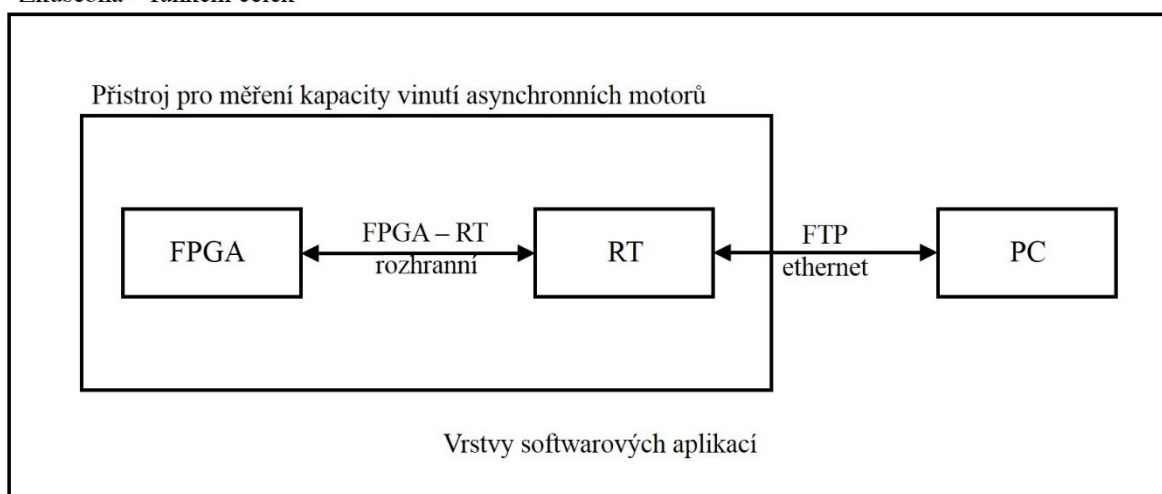
Nejprve jsem dostal za úkol oživit jednotlivé komponenty testeru zvlášť (komunikaci s displejem a čtečkou čárových kódů, seznámení se s vestavěným systémem, generování a vzorkování signálů atd.). Poté jsem začal pracovat na vytváření softwarových aplikací tří vrstev, které zmíněné komponenty spojují do funkčního celku, tedy výše zmíněného testeru.

První zadanou softwarovou aplikaci jsem měl programovat v LabVIEW a jeho přidruženém modulu pro programování hradlových polí FPGA (Field programmable gate array). Aplikace vykonává nízkoúrovňové úlohy v podobě zápisu a čtení vstupně-výstupních obvodů vestavěného systému a zasílání získaných dat do nadřazené aplikace.

Druhá zadaná softwarová aplikace stojící nad první aplikací byla programována opět v LabVIEW, ale s využitím přidruženého modulu pro programování procesorů s operačním systémem reálného času (Real-Time). Aplikace provádí řízení FPGA aplikace, diagnostiku měřených dat, práci s daty a soubory, komunikaci s periferiemi a synchronizaci dat s aplikací nadřazené zkušebny.

Poslední softwarová aplikace byla programována v základním vývojovém prostředí LabVIEW pro PC, který je součástí již stávající zkušebny realizované firmou ELCOM, a.s. Jednalo se spíše o softwarový modul, jehož funkcí byla synchronizace dat mezi vytvářeným testerem a firemní datovou sítí, dále také rozšířená diagnostika měřených dat.

Zkušebna – funkční celek



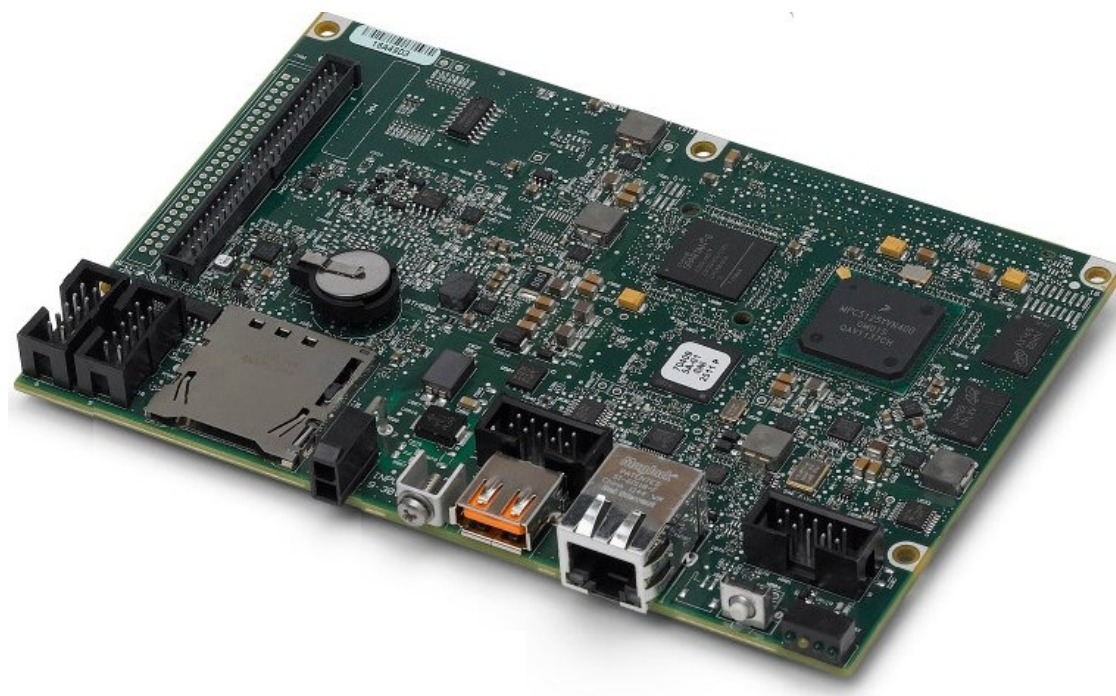
Obr. 3 Schéma uspořádání vytvářených softwarových vrstev

3 Postup řešení zadaných úkolů

V této kapitole budu postupně popisovat řešení jednotlivých úkolů zadání a také jakým způsobem se dílčí úkony a komponenty propojily do funkčního celku ve vytvořených softwarových aplikacích.

3.1 Seznámení se s vestavěným systémem a potřebným SW

Prvním zadaným úkolem bylo se seznámit s vestavěným systémem od firmy National Instruments Single-Board RIO 9626. Jedná se o komplexní integrovaný obvod, který v sobě kombinuje procesor s operačním systémem reálného času, programovatelné hradlové pole umožňující uživatelský přístup k analogovým a digitálním vstupně-výstupním obvodům. Dále se na kartě nachází celá řada komunikačních rozhraní, indikační LED a teplotní sensor. Systém tvoří hlavní část realizovaného testeru, jelikož s pomocí jeho zmíněných součástí a softwarových aplikací na něm spuštěných, jsou funkce všech ostatních použitých komponent propojeny do realizovaného testeru.



Obr. 4 Vestavěný systém NI sbRIO 9626 [3]

3.1.1 Technické parametry vestavěného systému NI sbRIO 9626

Podrobný popis celé konfigurace použitého vestavěného systému NI sbRIO 9626:

- Procesor 400 MHz s použitým RTOS (RTOS - Real time operating system) od firmy VxWorks.
- Xilinx Spartan-6 LX45 FPGA 40 MHz čip pro uživatelské nastavení ai/ ao a DIO.
- 512 MB pevná flash paměť pro ukládání programů a zápis dat.
- 256 MB DRAM pro deterministické vystavěné operace.
- 16 x16-bit multiplexovaných analogových vstupů s maximálním vzorkováním 200 kSa/s a 4 x 16-bit analogové výstupy.
- 4 x 3,3 V DIO linek a možnost rozšíření o dalších 96 DIO linek přes RMC konektor.
- Integrovaný 10/100 Mbit/s BASE-T ethernet port.
- 2 x RS232 a 1x RS485 sériové porty.
- 1x USB, CAN a SHDC port.
- Na desce se nachází konektor pro napájení v rozsahu 9-30 V.
- Vestavěný teplotní sensor, LED pro indikaci stavu zařízení a 1x uživatelsky využitelnou LED prostřednictvím FPGA.
- Tlačítko pro restartování zařízení popřípadě pro restartování IP adresy zařízení.

3.1.2 SW potřebný k nastavení a programování NI sbRIO 9626.

Pro programování a konfiguraci vestavěného systému sbRIO 9626 je potřebné mít nainstalované vývojové prostředí LabVIEW. Rozšiřující modul Real-Time (slouží pro programování procesorů s operačním systémem reálného času), modul FPGA (umožňuje programování hradlových bez využití programovacího jazyka VHDL) a NI-RIO (Reconfigurable input output) ovládače. Dále bylo nutné nainstalovat NI MAX (Measurement and automation explorer), s jehož pomocí je možné přistupovat ke všem produktům firmy National Instruments a konfigurovat je.

V dalším kroku jsem se seznámil s výše zmíněnými rozšiřujícími moduly vývojového prostředí LabVIEW a jejich rozdíly. Modul Real-Time obsahuje téměř všechny knihovny funkcí jako základní vývojové prostředí. Hlavní důraz při tvorbě programu bylo nutné klást na výkon a efektivitu software, protože použitý 400 MHz procesor má pouze omezený výkon. Modul FPGA obsahuje oproti základnímu vývojovému prostředí pouze knihovny pro základní logické a aritmetické operace, operace s poli a clustery, obsahuje navíc

knihovny pro přístup ke vstupně výstupním obvodům. Hradlové pole neumí pracovat s datovými typy s plovoucí desetinou čárkou, proto je možné dělení provádět pouze pomocí datového typu s pevně definovanou desetinou čárkou tzv. FXP (Fixed Point Data Type). Další nevýhodu programování v modulu FPGA je skutečnost, že každá funkce programu zabírá zdroje na hradlovém poli, takže je program omezen velikostí a je potřebné jej dobře rozvrhnout. Naopak obrovskou výhodou programování v modulu FPGA je možnost, využití tzv. skutečného paralelismu. Kód obsažený v každé smyčce využívá jiné zdroje a hradlové pole tak může vykonávat několik smyček najednou v různých časových intervalech, bez předávání zdrojů mezi smyčkami a procesy.

3.2 Generování excitačního signálu

Jako excitační signál pro měření kapacity impregnovaného svazku byl vybrán sinusový průběh o frekvenci 1 kHz, s nastavitelnou amplitudou v rozsahu od 0 do ± 10 V p-p. Pro generování excitačního signálu byl využit jeden z analogových výstupů, na který se pomocí aplikace vytvořené na hradlovém poli zapisují hodnoty požadovaného signálu. S ohledem k omezenému výkonu hradlového pole, jehož taktovací frekvence je 40 MHz, byla zvolena vzorkovací frekvence 250 kHz. Na jednu periodu generovaného signálu tedy připadá 250 vzorků.

Hodnoty generovaného průběhu byly vypočteny v programu Microsoft Excel dosazením do následujícího vzorce pro sinusový (harmonický) signál.

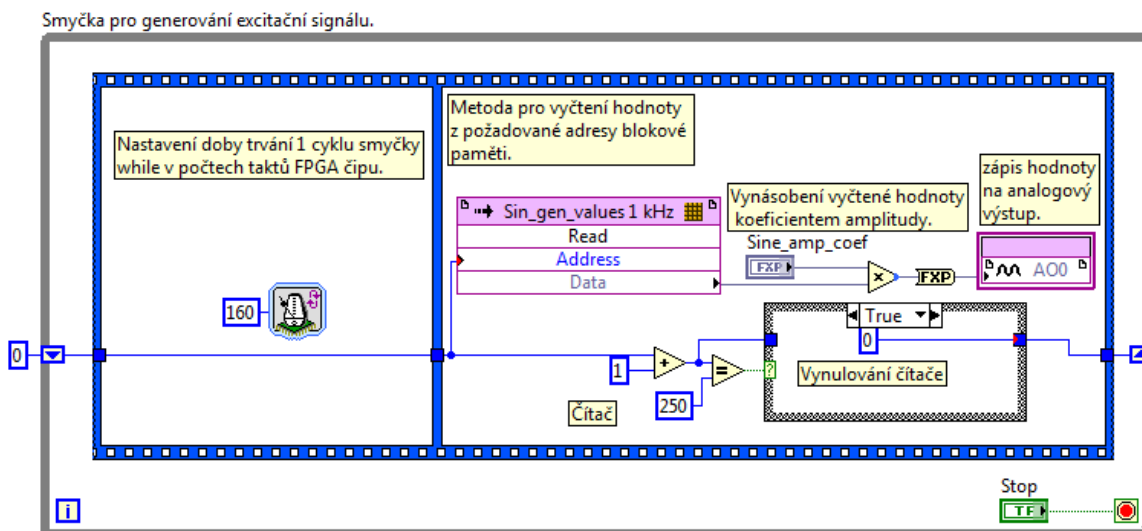
$$w(t) = W_m \cdot \sin(\omega \cdot t)$$

Kde $w(t)$ je okamžitá hodnota signálu v čase, W_m je amplituda napětí, ω je úhlová frekvence a t je okamžitá hodnota uplynulého času.

Při dosazení do vzorce byla zvolena amplituda signálu rovna 1, požadovaný rozsah amplitudy signálu se tak dal snadno získat násobením hodnoty vyčtené z paměti zvoleným koeficientem amplitudy (0-10).

Vypočtené hodnoty se uložily do blokové paměti hradlového pole, její velikost je určena počtem vzorků připadajících na 1 periodu, tedy 250. Datový typ prvků paměti byl zvolen shodně se vstupem funkce pro zápis hodnot na analogový výstup, tedy fixed point (FXP) s délkou slova 20 bitů a délkou celočíselného slova 5 bitů.

V programu se při každém cyklu smyčky, vybral požadovaný vzorek na odpovídající adrese v blokové paměti. Adresa vzorku je vybírána hodnotou čítače realizovaného pomocí zpětnovazebního uzlu. Velikost čítače odpovídá velikosti vytvořené blokové paměti. Jedno naplnění čítače tedy znamená vygenerování jedné periody excitačního signálu.



Obr. 5 Ukázka kódu, který slouží pro generování excitačního signálu

3.3 Oživení Čtečky čárových kódů

Pro projekt byla vybrána čtečka čárových kódů firmy SICK CLV 503, model patří do kategorie miniaturních čteček a byl vybrán, s ohledem na zákazníkův požadavek mobility přístroje. Čtečka využívá ke své funkci laserové diody, která emituje laserové záření o vlnové délce 650 nm (červená barva), záření spadá podle klasifikace do 2. třídy (laser operující v kontinuálním režimu ve viditelné části spektra s výkonem menším než 1 mW, ale větším než 400 μ W).

Model je vyráběn ve více provedeních podle typu komunikačního rozhraní, vzhledem k dostupným perifériím vestavěného systému jsme vybrali sériové komunikační rozhraní RS232.



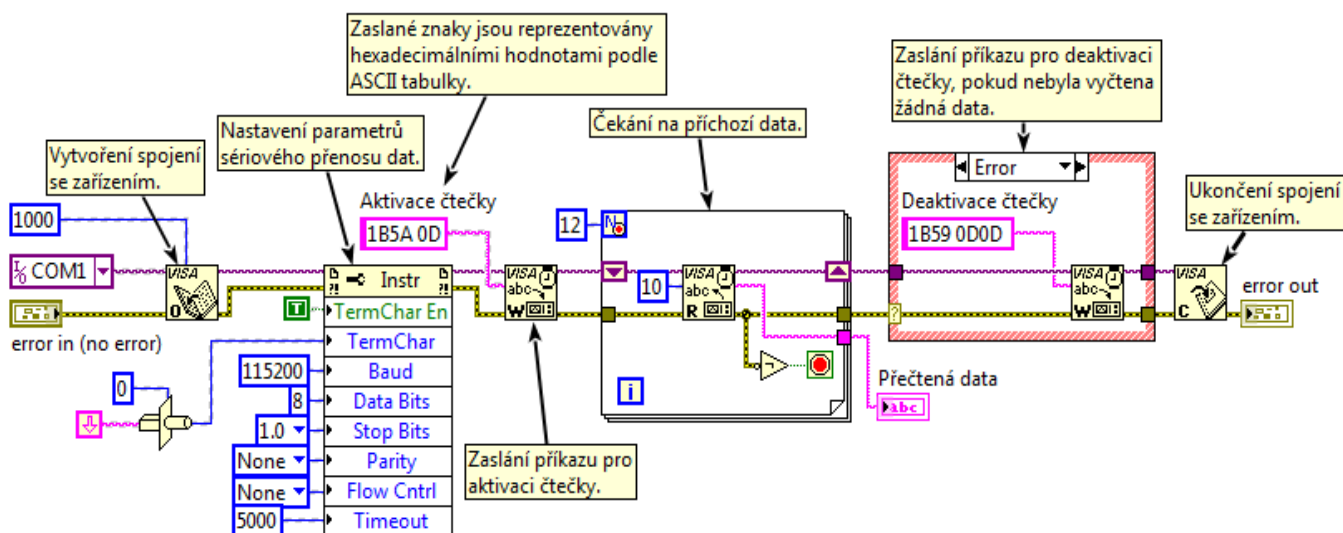
Obr. 6 Čtečka čárových kódů SICK CLV 503 [7]

3.3.1 Konfigurace datového přenosu a realizace komunikace

Před navázáním komunikace bylo nutné nastavit parametry sériového datového přenosu, nevolila se žádná kontrola toku dat, pouze se nastavila přenosová rychlost (115 200 baudů) počet datových bitů a počet stop bitů. Dále se nastavil pracovní režim zařízení, kdy čtečka po aktivaci čeká neomezeně dlouho na přečtení jednoho kódu. Jakmile čtečka přečte kód, odešle data a deaktivuje se.

Komunikace se čtečkou je velmi jednoduchá, v podstatě se jedná pouze o zaslání příkazů pro aktivaci a deaktivaci čtečky. Pokud je čtečka aktivní je potřeba neustále hlídat, jestli nezasílá informace o přečteném kódu.

Při programování komunikace se čtečkou čárových kódů využilo programovacího rozhraní NI VISA (Virtual instrument software architecture), které obsahuje knihovny pro komunikaci přes různá komunikační rozhraní. Komunikace byla programována v real-time modulu vývojového prostředí LabVIEW. Na obrázku níže je zobrazena ukázka kódu, který řeší komunikaci se čtečkou čárových kódů, pro přehlednost je kód doplněn o komentáře.



Obr. 7 Ukázka kódu pro komunikaci se čtečkou čárových kódů

3.4 Oživení Displeje

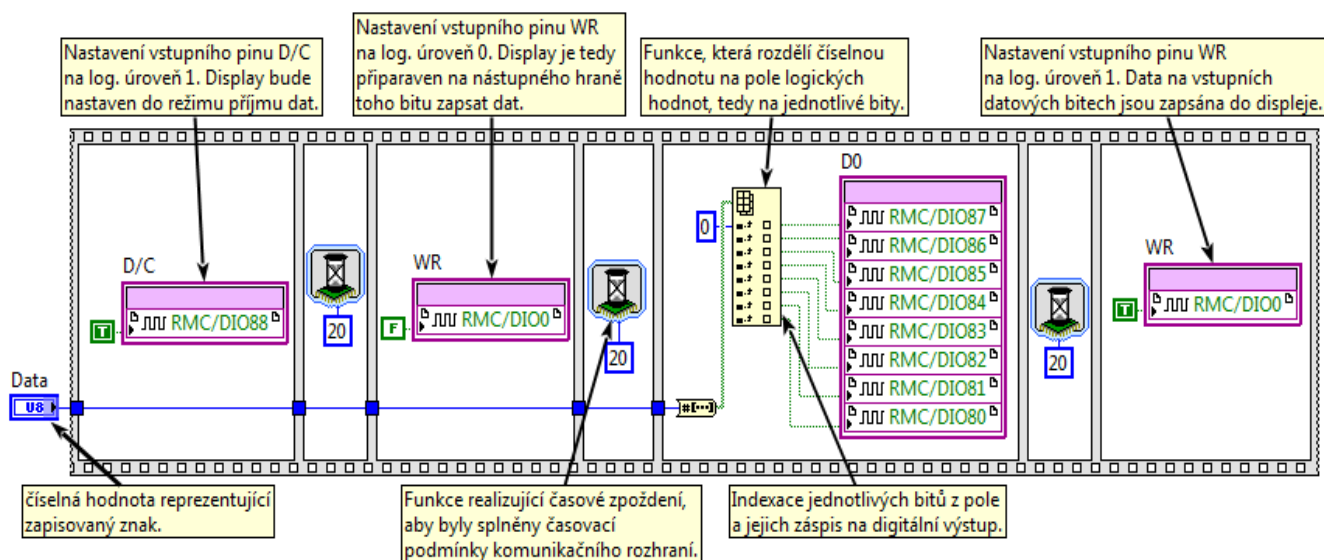
Při výběru displeje se opět kladl důraz na to, aby jeho rozměry a spotřeba byly co možná nejmenší. Pro tester byl nakonec vybrán plochý displej firmy Midas Components. Potřebám realizovaného testeru vyhovoval model MCOT42005AV-EWM. Jedná se o plochý OLED (Organic light-emitting diode) displej malých rozměrů s pasivní maticí (jednotlivé pixely jsou řízeny pasivně maticí navzájem překřížených vodičů). Displej disponuje 4 řádky, na každý řádek je možné zapsat až 20 znaků. Barva jednotlivých pixelů je pouze bílá, displej je tedy monochromatický (jednobarevný).



Obr. 8 Displej MCOT4500AV-EWM firmy MIDAS Components [8]

Při ožiování zvoleného displeje jsem úzce spolupracoval s kolegou z hardwarového oddělení, který vytvářel hardwarové komponenty umožňující propojení jednotlivých částí testeru. Oba jsme prostudovali technickou dokumentaci zvoleného displeje a jeho řídicí jednotky, kterou je čip SSD1311, tedy jednočipový mikroprocesor navržený pro řízení OLED. Podle technické dokumentace byla po naší vzájemné dohodě definována konfigurace displeje tak, aby odpovídala zvolenému komunikačnímu rozhraní Intel 8080, které využívá paralelního přenosu dat.

Poté jsem blíže prostudoval zvolené rozhraní pro komunikaci (datové bity, řídicí bity, časovací podmínky), sadu příkazů, se kterými OLED displej pracuje a doporučenou zapínací a vypínací sekvenci displeje. Podle těchto informací jsem vytvořil sadu funkcí (ovládačů) pro ovládání displeje s využitím výstupních digitálních linek hradlového pole. Na obr. 10 je možné vidět ukázkou kódu, jehož funkcí je zápis jednoho znaku na display, kód je doplněn o vysvětlující komentáře.



Obr. 9 Ukázkou kódu, který řeší zápis dat na displej

3.5 Sensor pro měření teploty

Realizovaný tester má sloužit k jednorázovému rychlému odměření kapacity asynchronních motorů po impregnaci. Jelikož je kapacita teplotně závislá je nutné změřit i okamžitou teplotu motoru v době měření. Okamžité měření teploty není možné realizovat standartními průmyslovými sensory (termočlánek, odporové teplotní detektory). Z tohoto důvodu byl vybrán bezkontaktní infračervený snímač teploty firmy MICRO-EPSILON řady ThermoMETER CS. Snímač umožňuje okamžité určení teploty povrchu tělesa měřením intenzity infračerveného záření, které těleso emituje.

Snímač obsahuje řídicí jednotku, kterou je možné konfigurovat softwarovou aplikací, poskytovanou výrobcem zařízení. Mým úkolem bylo sensor nakonfigurovat tak, aby poskytoval na svém výstupu napěťový signál v rozsahu 0 – 10 V odpovídající teplotnímu rozsahu 0 – 200 °C. Aby sensor správně určoval teplotu zkoumaného povrchu, bylo přes konfigurační software nutné zadat i emisivitu povrchu měřených motorů. Hodnota byla

určena na základě výsledků získaných měřením zkušebnímu modelu statoru motoru poskytnutého zákazníkem, později byla definována na skutečných motorech u zákazníka.

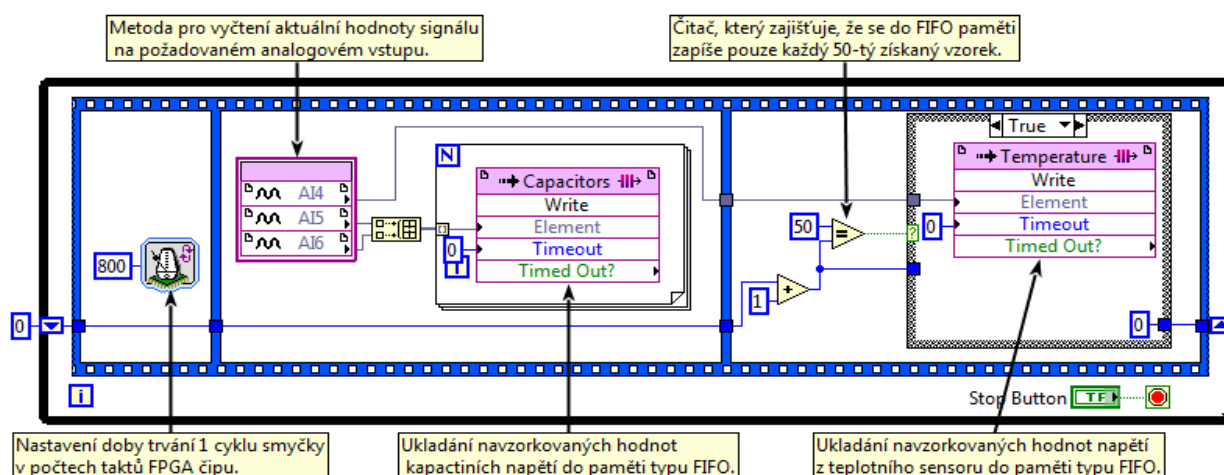


Obr. 10 Bezkontaktní infračervený snímač teploty firmy MICRO-EPSILON

3.6 Vzorkování měřených signálů

Posledním úkolem před programováním softwarových aplikací bylo vytvořit funkci, která bude vzorkovat měřené signály přivedené na analogové vstupy hradlového pole. Těmito signály jsou napětí na referenční kapacitě, napětí na kapacitě motoru a napěťový signál poskytovaný teplotním senzorem. Jelikož A/D převodník umožňuje vzorkovat pouze 200 kSa/s na všech vstupních kanálech dohromady (multiplexovaný vstup A/D převodníku), byla zvolena nejvyšší možná vzorkovací frekvence soudělná s taktovací frekvencí hradlového pole, tedy 50 kHz. Nejvyšší frekvence užitečného měřeného signálu je 1 kHz (frekvence generovaného excitačního signálu), zvolená vzorkovací frekvence tedy splňuje Nyquistův teorém a stačí také pro zachování tvaru měřených signálů.

Hradlové pole má pouze omezený výkon a nemá k dispozici téměř žádné pro analýzu signálu. Z tohoto důvodu se měřené vzorky okamžitě ukládají do paměti typu FIFO (first in first out), odkud se přenášejí do paměti, ke které má přístup procesor s operačním systémem reálného času. Tyto vzorky jsou poté v aplikaci mikroprocesoru vyčítány a analyzovány. Na obr. 12 je ukázka kódu, který řeší vyčítání dat z analogových vstupů a jejich zápis do pamětí typu FIFO. Pro přehlednost je kód opatřen komentáři.



Obr. 11 Ukázka kódu, který zajišťuje vzorkování signálů

3.7 Softwarová aplikace pro hradlové pole

Jak jsem již popsal v předchozí kapitole, tato aplikace zajišťuje přístup ke vstupně-výstupním obvodům vestavěného systému. Většinu důležitých funkcí použitých v této aplikaci jsem již vytvořil v přípravě a při ožívování komponent testeru. Při tvorbě aplikace jsem se soustředil hlavně na návrh správné softwarové architektury.

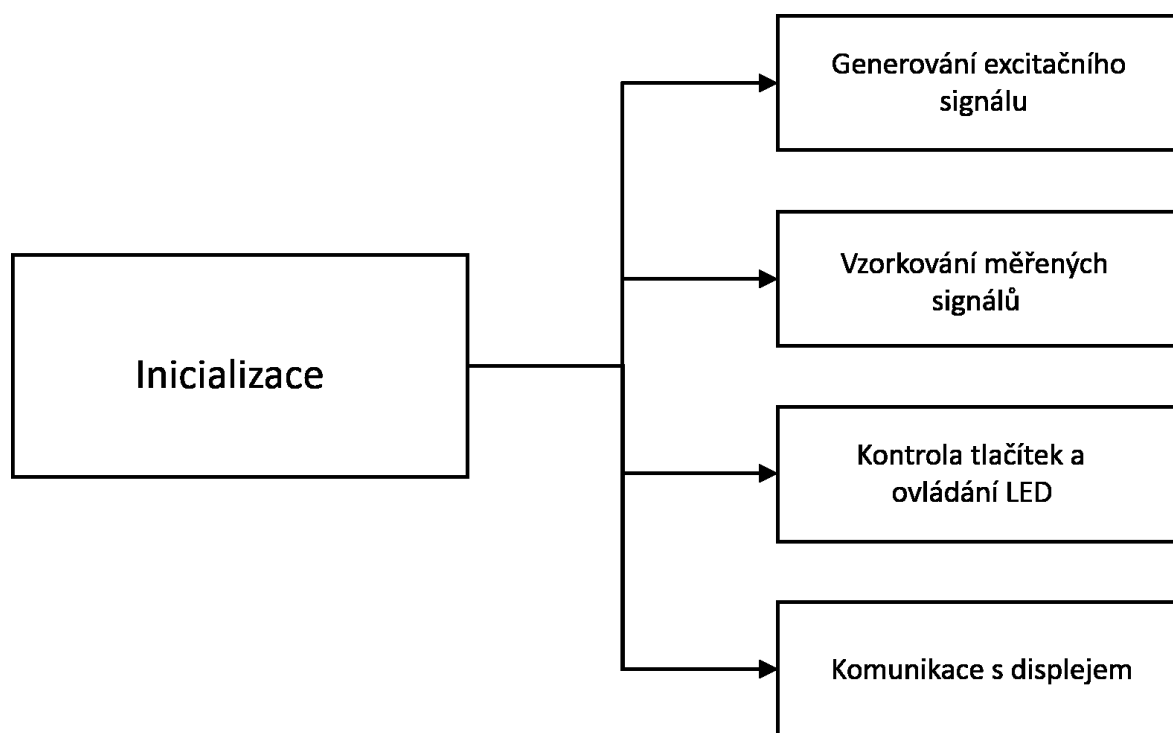
Aplikace je primárně rozdělena do dvou částí. V první části se provádí inicializace všech potřebných proměnných, spouštěcí sekvence displeje a poté se čeká na indikaci o připravenosti nadřazené RT aplikace. Během čekání se ve smyčce neustále aktualizují data na displeji, aby uživatel věděl, že přístroj pracuje. Při spuštění přístroje se totiž aplikace na hradlovém poli spustí téměř okamžitě, zatímco spouštění operačního systému reálného času na mikroprocesoru trvá až jednu minutu.

Ve druhé části aplikace jsem plně využil paralelismu hradlového pole a jeho přirozené podpory ve vývojovém prostředí LabVIEW. Po ukončení inicializace se v jednom okamžiku spouštějí čtyři smyčky, každá smyčka provádí úkony v pevně definovaných časových intervalech nezávisle na ostatních smyčkách. Důvodem pro spuštění smyček ve stejném okamžiku je synchronizace generování a vzorkování signálu, nemění se tak fáze vzorkovaného signálu v čase.

První dvě smyčky realizují generování excitačního signálu a vzorkování signálů měřených přístrojem. Popis jejich funkcí i s ukázkou použitého kódu jsem již rozebral v předchozích podkapitolách.

Třetí smyčka řeší ovládání indikačních LED a akustického prvku, pomocí časovacích a logických funkcí bylo vytvořeno několik režimů indikace, režim pro zvolenou LED je definován nadřazenou aplikací. Dále smyčka provádí záznam stisku tlačítek a opět s využitím časovacích a logických funkcí provádí filtraci záskmitů při stisku tlačítka a vytvoření příznaků o stisku tlačítka pro nadřazenou aplikaci.

Poslední smyčka hradlového pole vykonává neustálý zápis dat na displej z pole znaků. Každý prvek pole obsahuje jeden znak hlášení zapisovaného na displej. V této smyčce jsou použity funkce, jejichž tvorbu jsem zmínil v podkapitole 3.4. Na displej se data zapisují neustále, i když nedojde ke změně zapisovaných znaků, provádí se tím obnova znaků na displeji (vlivem rušení může dojít k přepsání znaků nebo k aktivaci nežádáných pixelů displeje).



Obr. 12 Schéma softwarové architektury aplikace hradlového pole

3.8 SW aplikace pro RTOS mikroprocesor

Pro tvorbu aplikace jsem využil stejné softwarové architektury jako u aplikace přechozí. Podstatným rozdílem však byla potřeba sdílet data mezi smyčkami a funkcemi, k tomu bylo využito lokálních a globálních proměnných. Musel jsem však zajistit, aby nedocházelo k souběhu při zapisování dat. Dále jsem oproti přechozí aplikaci použil ověřeného návrhového vzoru tzv. „stavového stroje“. Jedná se o strukturu, která ve smyčce vykonává určitý kód, jehož součástí je i rozhodovací algoritmus určující, jestli se kód vykoná znovu nebo přejde do jiného stavu. Použitím této softwarové architektury se mi podařilo docílit požadované sekvence vykonávání jednotlivých úloh. Vzniklý kód byl díky této volbě robustní, dobře čitelný a laditelný. V průběhu vývoje aplikace jsem řešil nebo se podílel na řešení níže popsanych problémů.

3.8.1 Přenos dat mezi aplikací hradlového pole

Zajištění přenosu dat mezi hradlovým polem a mikroprocesorem je v LabVIEW elegantně řešeno. Vytvořením reference na spuštěnou aplikaci hradlového pole lze využít jednoduchých funkcí pro přímý zápis do proměnných spuštěné aplikace hradlového pole. Stejně tak je možné aktuální hodnoty proměnných vyčítat. Tímto způsobem jsem z aplikace hradlového pole vyčítal příznaky o stisknutí jednotlivých tlačítek. A naopak jsem do ní zasílal příznaky pro aktivaci akustického prvku a indikačních LED. Dále jsem do aplikace zapisoval pole znaků obsahujících hlášení pro zobrazení na displej.

Pro přenos větších souborů dat má hradlové pole k dispozici paměti typu FIFO, které jsem využil k přenosu vzorků měřených signálů. K těmto pamětím je opět možné přistupovat přes připravené funkce a vyčítat tak celé soubory dat.

Hlavním problémem při přenosu dat mezi aplikacemi bylo zajištění správného časování, aby nedošlo ke ztrátě čtených nebo zapisovaných příznaků. Také bylo nutné synchronizovat rychlost vyčítání a vzorkování dat, jinak docházelo k přetečení paměti FIFO a ztrátě dat.

3.8.2 Vyhodnocení měřených dat

Před provedením vyhodnocení měření bylo nutné surová data měřených napět'ových signálů z referenční kapacity a měřeného motoru analyzovat a převést do podoby vhodné pro výpočet. Nejprve se vypočetla stejnosměrná složka signálů (střední hodnota signálu), která se od nich odečetla. Poté se určila efektivní hodnota signálů, jejich frekvence, celkové harmonické zkreslení THD (total harmonic distortion) a počáteční fáze. Z počátečních fází

se určil vzájemný fázový posun měřených signálů. Nakonec jsem použil efektivních hodnot signálů pro výpočet hrubé hodnoty kapacity zkoumaného motoru.

Pro určení jednotlivých parametrů signálů se využilo vestavěných funkcí vývojového prostředí LabVIEW. Tyto funkce k určení frekvence a fázového posuvu aplikují na zkoumaný soubor dat algoritmus rychlé fourierovy transformace FFT (fast fourier transform).

Ze signálu získaného teplotním senzorem se pouze určila stejnosměrná hodnota, která je přímo úměrná měřené teplotě. Jestliže teplota motoru přesáhla určitou mez, bylo nutné přepočíst hodnotu kapacity přes polynomickou funkci zadanou zákazníkem, která zohledňuje teplotní závislost kapacity motoru.

3.8.3 Kalibrace zařízení

Vyhodnocovacím algoritmem byla získána pouze hrubá (teoretická) hodnota kapacity motoru. Tato hodnota nezahrnuje chyby použité metody měření a chyby způsobené vlivem měřicího obvodu. Z tohoto důvodu se navrhl kalibrační postup, kterým se tyto chyby odstranily.

Nejprve se zvolila pracovní oblast testeru (rozsah měřených kapacit), poté se provedl náměr pomocí kapacitního etalonu (kapacitní dekády se známými hodnotami kapacit). Takto jsme získali hodnoty změřené přístrojem a k nim skutečné hodnoty pro zvolenou pracovní oblast. Nakonec jsem v aplikaci vytvořil algoritmus, který definuje koeficienty lineární závislosti, podle kterých se přepočítávají hodnoty získané přístrojem na skutečné hodnoty měřené kapacity. Pro zvýšení přesnosti měření algoritmus prokládá vždy dva vedlejší body získané při kalibračních náměrech.

3.8.4 Realizace algoritmů pro diagnostiku měřených dat

Součástí vyhodnocení měřených signálů je také diagnostika jejich parametrů. Aby bylo možné říct, že je změřená hodnota kapacity relevantní, musejí tyto parametry ležet v požadovaných intervalech. Součástí RT aplikace jsou tedy i algoritmy pro vyhodnocení efektivních hodnot, frekvencí, vzájemného fázového posuvu a celkového harmonického zkreslení signálů. Pokud některý z parametrů neodpovídal zadaným mezím, byla tato skutečnost indikována uživateli přístroje v podobě chybového hlášení, zvukové a světelné signalizace.

Diagnostické algoritmy také provádí kontrolu přítomnosti měřicího příslušenství přístroje a kontrolu správného spojení testeru s motorem v průběhu měření.

3.8.5 Realizace algoritmů pro práci se soubory a daty

Další nedílnou součástí aplikace bylo vytvoření algoritmů pro čtení a zápis dat do souborů, kontrolu dat získaných čtečkou čárových kódů a také přípravu měřených dat pro synchronizaci s aplikací nadřazené zkušebny. Seznámil jsem se se strukturou souborů používanou v oddělení zkušeben. Také jsem si osvojil práci s funkcemi pro úpravu textových řetězců a funkcemi pro zápis, čtení a hledání souborů, které jsou ve vývojovém prostředí LabVIEW dostupné.

3.8.6 Návrh uživatelského prostředí a ovládání přístroje

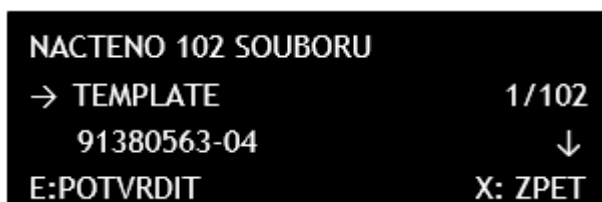
Před započítím realizace kódů řešících tvorbu textových hlášení pro zápis na displej a provedení úkonů přidružených ke stisku tlačítek jsem se podílel spolu s vedoucím projektu na návrhu uživatelské rozhraní a způsobu ovládání přístroje.

Prvním krokem bylo přidělení funkcí jednotlivým tlačítkům a LED přístroje s důrazem na ergonomii ovládání. Snahou bylo umožnit uživateli provádění základních úkonů pouze jednou rukou. A zajištění, aby všechny úkony s přístrojem zvládl pouze jeden uživatel bez asistence jiné osoby nebo nutnosti odložení přístroje (umožnění práce s přístrojem ve volném prostoru).



Obr. 13 Uživatelské rozhraní přístroje s popisy funkcí tlačítek a LED

Ve druhém kroku jsme se zabývali návrhem uživatelského menu a hlášení zobrazených na displeji s ohledem na předchozí přidělení funkcí jednotlivým tlačítkům. Navrhli jsme postupy přihlášení a odhlášení uživatele, načtení zakázek (měřených motorů) a postup měření. Ke každému postupu se také navrhla chybová a informativní hlášení spolu s vizuální a akustickou signalizací. Kvůli malým rozměrům displeje (omezený počet znaků) bylo největším úskalím vymyslet přehledný, jednoduchý a pro uživatele intuitivní způsob pohybu v uživatelském menu a tvorba seznamů.



Obr. 14 Ukázka uživatelského menu (seznam načtených zakázek)

3.8.7 Komunikace s jednotkou správy napájení

Komunikace s mikroprocesorem jednotky pro správu napájení byla realizována opět přes sériové rozhraní RS232. Na jejím vývoji jsem se podílel se zkušenějším kolegou z oddělení, přitom jsme spolupracovali s kolegy z hardwarového oddělení, kteří jednotku vyrobili a vytvářeli i software pro mikroprocesor.

Společně jsme navrhli jednoduchý komunikační protokol. Procesor řídící jednotky pro správu napájení při obdržení dotazovacího znaku od naší aplikace zaslal zpět status o stavu baterií (aktuální kapacita, doba do vybití resp. nabití baterie, indikace kritického stavu atd.). Při uživatelském požadavku na vypnutí přístroje se procesoru zaslal terminální znak, čímž se inicializovala vypínací sekvence.

3.8.8 Komunikace s aplikací nadřazené zkušebny

Komunikace probíhala přes rozhraní ethernet, bylo k ní využito sdílených proměnných vytvořených v nadřazené aplikaci. Tyto proměnné využívají protokoly PSP (publish and subscribe protocol) a TCP (transfer control protocol). Po spuštění aplikace, která je vytvořila, jsou dostupné pro čtení a zápis na síti.

Během trvání synchronizace dat testeru s nadřazenou zkušebnou, zasílala aplikace zkušebny informaci o tom, že synchronizace právě probíhá, spolu s indikací jejího průběhu. Aplikace zkušebny také zasílala příkaz testeru k odesílání surových průběhů měřených signálu, analyzovaných a diagnostikovaných dat.

3.9 SW aplikace pro PC nadřazené zkušebny

Kód této aplikace měl být vytvořen tak, aby mohl být spuštěn samostatně nebo být volán stávající aplikací zkušebny jako dynamická funkce. Při jejím vývoji jsem opět spolupracoval se zkušenějším kolegou z oddělení, který mi ukázal, jak této funkčnosti docílit. Aplikace primárně zajišťovala synchronizaci dat s realizovaným testerem, sekundárně měla aplikace poskytovat rozšířenou analýzu měřených dat. Její součástí byla také komunikace zmíněná v předchozí podkapitole.

Při vývoji bylo opět použito návrhového vzoru „stavový stroj“ pro zajištění sekvence úkonů. Protože součástí aplikace bylo i softwarové uživatelské rozhraní, získal jsem trochu zkušeností s jeho návrhem a také s používáním struktury pro správu uživatelských události v LabVIEW.

Synchronizace zahrnovala kopírování aktuálních souborů se vstupními daty pro měření (informace o měřených motorech) z datové sítě na realizovaný tester. Poté vymazání již starých souborů (měření již proběhlo, zakázka byla uzavřena) se vstupními daty z datového uložisti testeru. Nakonec se překopírovaly výstupní soubory s informacemi o provedeném měření z testeru do zákaznickovy datové sítě, při úspěšném kopírování se provedlo smazání souborů z testeru.

Rozšířenou analýzou dat bylo myšleno vytvoření uživatelského rozhraní pro přehlednou prezentaci (zobrazení osciloskopického průběhu, přehledné rozmístění indikátoru parametrů popsaných v podkapitolách 3.8.2 a 3.8.4) měřených dat zasílaných testerem. Rozhraní mělo sloužit pracovníkům firmy ELCOM, a.s. pro diagnostiku v případě poruchy přístroje.

3.10 Optimalizace a implementace vytvořeného softwaru

Vzhledem k omezeným zdrojům hradlového pole a výkonu mikroprocesoru byly aplikace v průběhu vývoje testovány z hlediska výkonu a plynulosti běhu. Určité původně použité funkce a části kódu musely být optimalizovány nebo nahrazeny (změna časování, použití blokových pamětí místo polí na FPGA, ořezání části měřeného signálu před analýzou atd.). Tato optimalizace byla nutná pro zajištění plynulosti v případě RT aplikace, a aby bylo dosaženo „vtlačení“ všech potřebných funkcí na dostupné zdroje hradlové pole.

Poslední část vývoje zahrnovala vytvoření souborů pro samostatné spuštění softwaru bez vývojového prostředí, jeho implementace do jednotlivých hardwarových platforem a nakonec otestování současného běhu všech aplikací na přístroji.

3.11 Předání přístroje zákazníkovi

Podílel jsem se také na tvorbě dokumentace k obsluze přístroje. Mým úkolem bylo vytvořit schémata zachycující postupy práce s přístrojem (přihlášení, odhlášení uživatele atd.) a popsat jednotlivá hlášení přístroje zobrazovaná na displeji.

Účastnil jsem se také předání přístroje zákazníkovi. Během předání se nejprve testovaly jednotlivé funkce přístroje v reálných podmínkách. Poté se provedla kalibrace přístroje za přítomnosti odborného zaměstnance zákazníka. Nakonec jsme spolu s kolegou zaškolili potenciální obsluhu realizovaného přístroje.



Obr. 15 Ukázka zkompletovaného přístroje pro měření kapacity

4 Teoretické a praktické znalosti získané studiem uplatněné v průběhu praxe

V průběhu praxe jsem nejvíce uplatnil své znalosti vývojového prostředí LabVIEW a také jiných softwarových produktů firmy National Instruments, které jsem získal studiem předmětů virtuální instrumentace. Přínosem pro mne bylo i studium předmětů spojených jinými vývojovými prostředími (MATLAB, Visual Studio, Eclipse).

Při oživování dílčích komponent přístroje jsem občas pracoval s nepájivým polem, kdy jsem využil znalosti z oblasti elektrických obvodů a techniky elektronických přístrojů. Během testování generování a vzorkování signálů a také při implementaci komunikace s displejem jsem uplatnil zkušenosti s používáním osciloskopu, digitálního multimetru a logického analyzátoru. Tyto zkušenosti jsem získal studiem předmětů technika elektronických přístrojů a měření na elektronických přístrojích.

Důležité pro mne byly i studiem nabyté znalosti matematiky a metody pro zpracování signálu, které jsem využil při analýze a vyhodnocení měřených dat

Velmi hojně jsem využil i znalosti anglického jazyka, hlavně technické terminologie, na kterou bylo studium soustředěno. S potřebou jazyka jsem se setkal při studování dokumentace jednotlivých komponent přístroje, která byla k dispozici většinou pouze v anglickém jazyce. A také při práci s nápovědou vývojového prostředí LabVIEW.

5 Scházející znalosti a dovednosti v průběhu praxe

Během praxe mi scházely znalosti silnoprůdých zařízení, hlavně v oblasti indukčních motorů, pro jejichž měření byl přístroj realizován. Všechny potřebné informace mi však byly při realizaci přístroje poskytnuty, takže mi tato skutečnost nebránila při vykonávání zadaných úkolů.

Při realizaci úkolů mi chyběly zkušenosti s programováním hradlových polí. Pomocí rozšiřujícího modulu FPGA vývojového prostředí LabVIEW je však možné programovat hradlové pole intuitivně, bez odborných znalostí této problematiky. Studium dostupných podpůrných výukových materiálů poskytovaných firmou National Instruments jsem získal základní informace o principech funkce hradlového pole, ze kterých plynou jeho výhody i nevýhody. Následně jsem byl schopen využít jeho potenciál, implementovat postupy obecné algoritmizace a vyhnout se tak nejčastěji prováděným chybám při programování.

Scházel mi také zkušenosti s komunikací mezi zařízeními. Využitím znalostí získaných v průběhu studia spolu s metodou pokus a omyl jsem se však vždy dobral zdárného výsledku. Získal jsem tak zkušenosti s paralelním i sériovým přenosem dat a různými komunikačními rozhraními a protokoly.

6 Závěr a zhodnocení praxe

Z mého pohledu shledávám absolvovanou praxi jako zajímavou a velmi přínosnou. V průběhu praxe jsem se seznámil s konceptem přístroje pro měření kapacity impregnovaného svazku asynchronních motorů. Oživil jsem jeho jednotlivé komponenty. Samostatně nebo s malou pomocí kolegů z oddělení jsem navrhl a implementoval SW aplikace pro ovládání přístroje. Nakonec jsem se účastnil testování přístroje a jeho předání zákazníkovi. Všechny zadané úkoly se mi podařily splnit ve stanoveném termínu.

Během absolvování praxe jsem se seznámil s filozofií a způsobem fungování firmy. Měl jsem možnost účastnit se výrobních procesů a návrhu produktu podle zadání zákazníka. Získal jsem povědomí o náročnosti a důležitosti výrobních postupů nutných pro zajištění úspěšné realizace spolehlivého zařízení. A také o náročnosti ekonomických a technických podmínek při vývoji nového zařízení pro průmysl.

Nedílnou součástí praxe byla kolektivní spolupráce, komunikace s kolegy se stejným odborným zaměřením, ale také s kolegy, kteří se zabývají návrhem a tvorbou HW komponent. Díky této zkušenosti jsem si uvědomil význam a přínos kooperace při vývoji komplexního zařízení. Dále jsem měl možnost si vyzkoušet komunikaci se zákazníkem, a to zejména při předávání přístroje a zaškolování jeho budoucí obsluhy.

Při vývoji SW aplikací pro realizované zařízení jsem pracoval s moderními technickými prostředky a produkty celosvětově uznávaných firem jako National Instruments, SICK a jiné. Dále jsem využil příležitostí podívat se na produkty i jiných firem, se kterými pracovali kolegové z různých oddělení divize.

Pracovní náplní vykonané odborné praxe byl vývoj aplikací k zařízení určenému pro silnoproudý elektrotechnický průmysl. Přesto je možné uplatnit praxí nabyté technické dovednosti a znalosti k vývoji zařízení pro zdravotnický průmysl. Především zkušenosti v oblasti měření, vyhodnocení, přenosu dat a také s návrhem a implementací SW ve vývojovém prostředí LabVIEW.

Seznam použité literatury a zdrojů

- [1] ELCOM, a.s. O Společnosti. *ELCOM, a.s.* [online]. 2015 [cit. 2015-01-18]. Dostupné z: <http://www.elcom.cz/cz/spolecnost/o-spolecnosti>
- [2] ELCOM, a.s. Divize. *ELCOM, a.s.* [online]. 2015 [cit. 2015-01-18]. Dostupné z: <http://www.elcom.cz/cz/spolecnost/divize>
- [3] NI sbRIO-9626. *National Instruments Corporation* [online]. 2014 [cit. 2015-01-18]. Dostupné z: <http://www.sine.ni.com/nips/cds/view/p/lang/cs/nid/210419>
- [4] *OEM OPERATING INSTRUCTIONS AND SPECIFICATIONS: NI sbRIO-9605/9606 and NI sbRIO-9623/9626/9633/9636* [Online]. 2014 [cit. 2015-01-18]. ISBN 373378D-01. Dostupné z: <http://digital.ni.com/manuals.nsf/websearch/E065536DC595BDD886257C2E00704610>
- [5] *CLV503 / CLV505 Bar Code Scanners: Quick start* [online]. 2009 [cit. 2015-01-18]. ISBN 8012892. Dostupné z: http://wwwnew.sickcn.com/media/3450/QSCLV503_505DEEN_8012892_200902.pdf
- [6] Bar code scanners: CLV50x / CLV503 / Mid Range. *SICK Sensor Intelligence* [online]. 2015 [cit. 2015-01-18]. Dostupné z: <https://www.sick.com/de/en/identification-solutions/bar-code-scanners/clv50x/clv503-0000/p/p126440>
- [7] New Midas White Passive matrix OLED Display. *RS Components* [online]. 2015 [cit. 2015-01-18]. Dostupné z: <http://sg.rs-online.com/web/p/oled-displays/8195156/>
- [8] *Specification: MCOT42005AV-EWM* [online]. 2013 [cit. 2015-01-18]. Dostupné z: <http://www.midasdisplays.com/pdf/MCOT42005AX.pdf>
- [9] *SSD1311: Advance Information OLED/PLED Segment / Common Driver with Controller* [online]. 2011 [cit. 2015-01-18]. Dostupné z: <http://www.solomon-systech.com/en/product/display-ic/oled-driver-controller/ssd1311/>
- [10] MICRO-EPSILON. *Instruction Manual thermoMETER CS* [Online]. 2008, 38 s. [cit. 13. 4. 2015]. Dostupné z: <http://www.micro-epsilon.com/download/manuals/man--thermoMETER-CS--en.pdf>

Seznam obrázků

Obr. 1 Budova divize virtuální instrumentace firmy ELCOM, a.s. [2]	12
Obr. 2 Přehledové schéma realizovaného testeru	14
Obr. 3 Schéma uspořádání vytvářených softwarových vrstev	15
Obr. 4 Vestavěný systém NI sbRIO 9626 [3]	16
Obr. 5 Ukázka kódu, který slouží pro generování excitačního signálu	19
Obr. 6 Čtečka čárových kódů SICK CLV 503 [7]	20
Obr. 7 Ukázka kódu pro komunikaci se čtečkou čárových kódů	21
Obr. 8 Displej MCOT4500AV-EWM firmy MIDAS Components [8]	21
Obr. 9 Ukázka kódu, který řeší zápis dat na displej	22
Obr. 10 Bezkontaktní infračervený snímač teploty firmy MICRO-EPSILON	23
Obr. 11 Ukázka kódu, který zajišťuje vzorkování signálů	24
Obr. 12 Schéma softwarové architektury aplikace hradlového pole	25
Obr. 13 Uživatelské rozhraní přístroje s popisy funkcí tlačítek a LED	28
Obr. 14 Ukázka uživatelského menu (seznam načtených zakázek)	29
Obr. 15 Ukázka zkompletovaného přístroje pro měření kapacity	31